

CLAIMS

I claim:

5 1. A method for enabling a first software program
 using a first binary specification in a first execution
 environment to employ a limited functionality of a
 second software program using a second binary
 specification in a second execution environment, the
10 method comprising:
 creating a bridge in said first execution
 environment; and
 creating, in said first execution environment
 using said bridge, a proxy wrapping an interface
15 to said limited functionality of said second
 software program in said second execution
 environment.

20 2. A method as in Claim 1 further comprising:
 creating a first execution environment object
 including said second binary specification.

25 3. A method as in Claim 2 further comprising:
 creating a second execution environment
 object including said first binary specification.

30 4. A method comprising:
 generating a binary specification object for
 a first execution environment;
 generating a binary specification object for
 a second execution environment; and
 generating a bridge object wherein said
 bridge object is used in mapping objects from said
 second execution environment to said first
35 execution environment.

5. The method of Claim 4 further comprising:
 using said bridge object to generate a proxy
 wrapping an interface in said second execution
 environment.

5

6. A method for using functionality in a second
execution environment in a first execution environment
comprising:

10

 calling a method in a proxy interface in said
 first execution environment; and
 converting said method call by said proxy
 interface to a corresponding method call for
 execution in said second execution environment.

15

7. The method as in Claim 6 further comprising:
 dispatching said corresponding method call
 for execution in said second execution environment
 to said second execution environment by said proxy
 interface.

20

8. The method of Claim 6 where said converting
said method call further comprises:
 using a type description to convert
 parameters from said first execution environment
 to said second execution environment.

25

9. The method of Claim 7 further comprising:
 executing said corresponding method call in
 said second execution environment, and returning
 results of said execution to said proxy interface.

30

10. The method of Claim 9 further comprising:
 using a type description to convert said
 returned results from said second execution
 environment to said first execution environment.

35

05260324.011201

11. The method of Claim 6 wherein said second execution environment is a C++ programming language execution environment.

5 12. A method for using functionality in a second execution environment in a first execution environment comprising:

calling a method in a proxy interface in said first execution environment;

10 converting said method call by said proxy interface to a corresponding method call for execution in said second execution environment, wherein said converting said method call comprises:

15 using a type description to convert parameters from said first execution environment to said second execution environment; and

20 dispatching said corresponding method call for execution in said second execution environment to said second execution environment by said proxy interface.

13. The method of Claim 12 further comprising:

25 executing said corresponding method call in said second execution environment, and returning results of said execution to said proxy interface.

14. The method of Claim 13 further comprising:

30 using a type description to convert said returned results from said second execution environment to said first execution environment.

15. A computer program product comprising
35 computer program code for a method for enabling a first software program using a first binary specification in

a first execution environment to employ a limited functionality of a second software program using a second binary specification in a second execution environment, the method comprising:

5 creating a bridge in said first execution environment; and

 creating, in said first execution environment using said bridge, a proxy wrapping an interface to said limited functionality of said second 10 software program in said second execution environment.

16. The computer program product of Claim 15 wherein said method further comprises:

15 creating a first execution environment object including said second binary specification.

17. The computer program product of Claim 16 wherein said method further comprises:

20 creating a second execution environment object including said first binary specification.

18. A computer program product comprising computer program code for a method for using 25 functionality in a second execution environment in a first execution environment, said method comprising:

 calling a method in a proxy interface in said first execution environment; and
 converting said method call by said proxy 30 interface to a corresponding method call for execution in said second execution environment.

19. The computer program product of Claim 18 wherein said method further comprises:

35 dispatching said corresponding method call for execution in said second execution environment

to said second execution environment by said proxy interface.

20. The computer program product of Claim 18
5 wherein said method further comprises:

using a type description to convert
parameters from said first execution environment
to said second execution environment.

10 21. The computer program product of Claim 19
wherein said method further comprises:

executing said corresponding method call in
said second execution environment, and returning
results of said execution to said proxy interface.

15 22. The computer program product of Claim 21
wherein said method further comprises:

using a type description to convert said
returned results from said second execution
environment to said first execution environment.

20 23. A computer storage medium having stored
therein a structure comprising:

25 a binary specification for an execution
environment including:
a simple common identity structure.

24. The computer storage medium of Claim 23
wherein said binary specification further comprises:

30 an extended environment structure.

25. The computer storage medium of Claim 23
wherein said simple common identity structure includes:
a type name.

35

26. The computer storage medium of Claim 23
wherein said simple common identity structure includes:
a method acquire.

5 27. The computer storage medium of Claim 23
wherein said simple common identity structure includes:
a method release.

28. The computer storage medium of Claim 24
10 wherein said simple common identity structure includes:
a pointer to said extended environment
structure.

09760321.014201